



\*\*FILE\*\*ID\*\*DAPDEF

K 14

DDDDDDDD DDDDDDDDD  
DDDDDDDD DDDDDDDDD  
DD DD AA AA PP PP DD EE FF  
DD DD AA AA PP PP DD EE FF  
DD DD AA AA PP PP DD EE FF  
DD DD AA AA PPPPPPPP DD DD EEEEEEEE FFFFFFFF  
DD DD AA AA PPPPPPPP DD DD EEEEEEEE FFFFFFFF  
DD DD AAAAAAAA PP DD DD EE FF  
DD DD AAAAAAAA PP DD DD EE FF  
DD DD AA AA PP DD DD EE FF  
DD DD AA AA PP DD DD EE FF  
DDDDDDDD AA AA PP DDDDDDDDD EEEEEEEE FF  
DDDDDDDD AA AA PP DDDDDDDDD EEEEEEEE FF  
.....  
.....  
.....  
.....

MM MM DDDDDDDDD LL  
MM MM DDDDDDDDD LL  
MM MM DD DD LL  
MM MM DDDDDDDDD LLLLLLLLLL  
MM MM DDDDDDDDD LLLLLLLLLL

:TITLE \$DAPDEF - DATA ACCESS PROTOCOL DEFINITIONS  
:IDENT 'V04-000'

\*\*\*\*\*  
\* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
\* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
\* ALL RIGHTS RESERVED.

\*\*\*\*\*  
\* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
\* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
\* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
\* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
\* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
\* TRANSFERRED.

\*\*\*\*\*  
\* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
\* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
\* CORPORATION.

\*\*\*\*\*  
\* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
\* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

++ Facility: DAP (Data Access Protocol)

Abstract:

This module defines the DAP control block. It is both an input and output control structure for the FAL\$DECODE\_MSG and NT\$DECODE\_MSG subroutines in FAL and RMS, respectively.

Environment:

The MDL translator must be used to convert DAPDEF.MDL into DAPDEF.MAR (and DAPDEF.B32).

Author: James A. Krycka. Creation Date: 17-OCT-1977

Modified By:

V03-007 JEJ0018 J E Johnson 27-Mar-1984  
Correct double assignment of DAP\$V\_POS caused in V03-006; now  
P/OS will be identified as DAP\$V\_P\_OS and DAP\$K\_P\_OS.  
Remove no longer used DAP buffer size constants:  
DAP\$K\_INIBUFSIZ, DAP\$K\_MINBUFSIZ, and DAP\$K\_MAXBUFSIZ.

V03-006 JAK0124 J A Krycka 06-SEP-1983  
Define operating system class bits analogous to DAP\$V\_VAXVMS  
(VAXELAN, TOPS10, TOPS20, RT11, RSTS, RSX, IAS, and POS).  
Define DAP\$B\_X\_FIELD containing flags from DAP\$Q\_DCODE\_FLG.

Rearrange order of DAP\$Q\_DCODE\_FLG bits.

V03-005 JAK0112 J A Krycka 22-JUN-1983  
Define DAP\$V\_GEQ\_V71.  
Define DAP\$V\_VMS\_XPF1 thru VMS\_XPF4.

V03-004 JAK0111 J A Krycka 17-JUN-1983  
Upgrade definitions to correspond to DAP V7.0 specification:  
Define DAP\$K\_VAXELAN and DAP\$K\_RMS32S.  
Define new SYSCAP bit (OCTALVER).  
Define DAP\$K\_IN8 and DAP\$K\_BN8.

V03-003 KRM0102 K Malik 09-May-1983  
Define new SYSCAP field bits (MODATTCRE, NAM3PART, CHGATTRN,  
CHGTIMREN, CHGPROREN, BLK(NT)).  
Rename SYSCAP bits (CHGATT to CHGATTCLS, CHGTIM to CHGTIMCLS,  
CHGPRO to CHGPROCLS, CHGNAM to CHGNAMCLS).  
Define DAP\$V\_BLKCNT, DAP\$B\_BLKCNT, and DAPS\_BLKCNT.  
Define DAP\$V\_DSP\_3NAM.  
Define DAP\$K\_QUIT.

V03-002 KRM0081 K Malik 23-Mar-1983  
Define DAP\$V\_GEQ\_V70.  
Rename DAP\$B\_SOFTVER to DAP\$B\_DECVER.  
Rename DAP\$B\_USRSOFT to DAP\$K\_USRVER.  
Define DAP\$K\_STMLF and DAP\$K\_STMCR.

V03-001 KRM0065 K Malik 23-NOV-1982  
Change DAP\$K\_SYSCAP2\_V and DAP\$K\_VALID\_R2F values to support  
rename operation.

V02-047 JAK0070 J A Krycka 27-JAN-1982  
Remove all "DAP\$V\_..." symbols from expressions and eliminate  
the use of ":" in symbol names to aid in future conversion of  
this MDL file into SDL format.

V02-046 JAK0063 J A Krycka 24-AUG-1981  
Cleanup:  
Rearrange sections defined by \$DAPPLGDEF.  
Expand several menu fields from one byte to two bytes in length  
(DAP\$W\_CTLMENU, DAP\$W\_TIMENU, DAP\$W\_PROMENU, DAP\$W\_SUMENU).  
For consistency, denote fields that exist in two messages as  
DAP\$S\_name1 and DAP\$S\_name2 (FOP, ALQ, DEQ, DISPLAY, RECNUM).  
Remove unused system specific fields (DAP\$L\_FOP, DAP\$L\_ROP, and  
DAP\$L\_CTX).  
Rename SYSCAP bits (RANREC to RANRRN, MULKEY to IDXORG, and  
BITCOUNT to BITOPT).

V02-045 JAK0063 J A Krycka 21-AUG-1981  
Upgrade definitions to correspond to DAP V6.0 specification:  
Define DAP\$V\_GEQ\_V60.  
Define DAP\$V\_EXTEND and DAP\$V\_DISPLAY.  
Define new SYSCAP field bits TEXTEND, DISPLAY, GNGOPT, CHGATT,  
CHGTIM, CHGPRO, and CHGNAM).  
Define new FOP field bit (DIR).  
Define new ROP field bits (ROPWAT, RRL, and REA).

Define DAPSK\_EXTEND\_B and DAPSK\_EXTEND\_E; remove DAPSK\_EXTEND.  
Define DAPSK\_CHANGE\_B, DAPSK\_CHANGE\_E, and DAPSK\_TERMINATE.  
Rename DAPSK\_PURGE to DAPSK\_RESET.  
Define DAPSQ\_STX and DAPS\_STX.  
Define DAPSV\_PDT, DAPSQ\_PDT, and DAPS\_PDT.  
Define DAPSV\_ADT, DAPSQ\_ADT, and DAPS\_ADT.  
Modify value of DAPSK\_SYSACP1\_V (set EXTEND and DISPLAY bits).  
Modify value of DAPSK\_SYSACP2\_V (set CHGTIM and CHGPRO bits).

V02-044 JAK0061 J A Krycka 17-JUL-1981  
Define DAPSK\_INIBUFSIZ, DAPSK\_MINBUFSIZ, and DAPSK\_MAXBUFSIZ.  
Remove DAPSK\_BUFSIZ\_F and DAPSK\_BUFSIZ\_R.

V02-043 JAK0060 J A Krycka 23-JUN-1981  
Define DAPSK\_TOPS10, and DAPSK\_TOPS10F.  
Define DAPSV\_BDT, DAPSQ\_BDT, and DAPS\_BDT.  
Modify value of DAPSK\_Flags\_U (remove LEN256 bit).  
Modify value of DAPSK\_SYSACP1\_V (set RANRFA and BIGBLK bits).

V02-042 JAK0050 J A Krycka 22-NOV-1980  
Define DAPSV\_RMS and DAPSV\_FCS.  
Fix bug in definition of reserved bit in FOP field.  
Change DAPSK\_BUFSIZ\_F value from <4096+256> to <4096+32>.  
Modify value of DAPSK\_SYSACP2\_V (include WILDCARD bit).

V02-041 REFORMAT J A Krycka 26-JUL-1980

```

++ Define the overall structure of the DAP control block and symbols related
to its prologue section.

Note: Longword and quadword fields are longword aligned within the control
block. Fields longer than 8 bytes are not stored within. Instead a
descriptor is stored in the control block that points to an external
buffer where the field data is located.

SSTRUCT DAP,PLGDEF      : DAP control block prologue
:
: Parameter and status section
:
F DCODE_FLG,Q
:
S VERSION,4,W
S PARTNER,6,W
V <
  ,32
  GEO_V41
  GEO_V42
  GEO_V52
  GEO_V54
  GEO_V56
  GEO_V60
  GEO_V70
  GEO_V71
  ,4
  VMS_XPF1
  VMS_XPF2
  VMS_XPF3
  VMS_XPF4
  RMS
  FCS
  STM_ONLY
  ,1
  VAXVMS
  VAXELAN
  TOPS10
  TOPS20
  RT11
  RSTS
  RSX
  IAS
  P_OS
  ,3
  >
F MSG_BUF1,Q
F MSG_BUF2,Q
:
: Message decode status flags
: (output from message decode subroutine)
: Note: bits 00-31 are defined external to DAP
: Note: bits 32-63 are defined by DAP herein
: Remote DAP protocol version level (bits 32-47)
: Remote system classification (bits 48-63)
: Meaning:
: Skip over reserved bits
: Partner implemented to DAP since V4.1
: Partner implemented to DAP since V4.2
: Partner implemented to DAP since V5.2
: Partner implemented to DAP since V5.4
: Partner implemented to DAP since V5.6
: Partner implemented to DAP since V6.0
: Partner implemented to DAP since V7.0
: Partner implemented to DAP since V7.1
: Spare
: VAX/VMS experimental protocol option flag
: Partner uses an RMS based file system
: Partner uses an FCS based file system
: Partner uses a stream ASCII based file system
: Spare
: Partner runs under VAX/VMS
: Partner runs under VAXELAN
: Partner runs under TOPS-10
: Partner runs under TOPS-20
: Partner runs under RT-11
: Partner runs under RSTS/E
: Partner runs under RSX-11M, -11MP, or -11S
: Partner runs under IAS or RSX-11D
: Partner runs under PO/S
: Spare
: On input, descriptor of message string
: to decode
: On output, descriptor of string remaining
: after message just decoded
: On input, ignored

```

; On output, descriptor of message just  
 ; decoded; same as MSG\_BUG1 on input if  
 ; no blocked message follows  
 Message decode status codes  
 (output from message decode subroutine)  
 Message decode success/fail (1/0) status flag  
 On error, DAP field ID code; else 0  
 Message type (0 if invalid)  
 On error, DAP MACCODE error code; else 0  
 Bit mask of valid messages to receive  
 (input to message decode subroutine)  
 (bit offsets are derived from message type  
 values, e.g., offset for Data message is  
 <1@DAP\$K DAT\_MSG>)  
 Current CRC resultant value  
 Explicit field found in message flags field  
 Meaning:  
 Message explicitly contained RECNUM field  
 Message explicitly contained CHECK field  
 Spare  
 Spare  
 -----
 Message decode section (part 1)  
 -----
 Configuration message save section  
 (space for DAP\$0 SYSCAP bit mask field  
 defined by the \$DAPCNFDEF macro)  
 -----
 Message decode section (parts 2 and 3)  
 -----
 Current message work area  
 Current message work area size  
 Message header section  
 (space for current message header fields  
 defined by the \$DAPHDRDEF macro)  
 Message operand section  
 (space for current message operand fields  
 defined by the \$DAPxxxDEF macros, where xxx  
 represents the 15 DAP message mnemonics)  
 \*\*\*\*\* offset = ^X80 = 128 \*\*\*\*\*  
 -----
 Message decode section (parts 4 and 5)  
 -----
 System specific work area  
 System specific work area size  
 System specific section  
 (space for system specific fields  
 defined by the \$DAPSSPDEF macro)  
 Temporary work area  
 Temporary work area size  
 Spare  
 -----
 Define length of DAP control block

;++  
; Define symbols related to the DAP message header.  
;--

```
$STRUCT DAP,HDRDEF      ; DAP message header
F .L.12                  ; Position to message header section
                           ; of DAP control block
F TYPE,B                 ; DAP message type field (1) : B
  K <
    CNF_MSG,1             ; DAP message type:
    ATT_MSG,2              ; Configuration message
    ACC_MSG,3              ; Attributes message
    CTL_MSG,4              ; Access message
    CON_MSG,5              ; Control message
    CON_MSG,5              ; Continue Transfer message
    ACK_MSG,6              ; Acknowledge message
    CMP_MSG,7              ; Access Complete message
    DAT_MSG,8              ; Data message
    STS_MSG,9              ; Status message
    KEY_MSG,10             ; Key Definition Attributes message
    ALL_MSG,11             ; Allocation Attributes message
    SUM_MSG,12             ; Summary Attributes message
    TIM_MSG,13             ; Date and Time Attributes message
    PRO_MSG,14             ; Protection Attributes message
    NAM_MSG,15             ; Name message
    >                      ; (16) reserved for ACL Attributes message
  K VALID R2F,-           ; Mask of DAP messages valid for RMS to send:
    <"X0000EDBE>          ; CNF, ATT, ACC, CTL, CON, CMP, DAT, KEY, ALL,
                           ; TIM, PRO, NAM
  K VALID F2R,-           ; Mask of DAP messages valid for FAL to send:
    <"X0000OFFC6>          ; CNF, ATT, ACK, CMP, DAT, STS, KEY, ALL, SUM,
                           ; TIM, PRO, NAM
F FLAGS,B                ; DAP message flags field (EX-5) : BM
  V <M
    STREAMID              ; Menu of fields to follow:
    LENGTH
    LEN256
    BITCNT
    TMP1$,1
    SYSPEC
    SEGMENT
    TMP2$,1
  >
  K FLAGS I,<-            ; Define flags options that are invalid:
    <DAP$R_TMP1$>!-        ; Reserved
    <DAP$M_TMP2$>!-        ; Reserved
    0>
  K FLAGS U,<-            ; Define flags options unsupported by VAX:
    <DAP$R_BITCNT>!-        ; BITCNT
    <DAP$M_SEGMENT>!-        ; SEGMENT
    0>
F STREAMID,B             ; Data stream identification field (1) : B
F LENGTH,B               ; Length (of rest of message) field (1) : B
```

```
F LEN256,B      ; Length extension field (1) : B
F BITCNT,B     ; Bit count field (1) : B
F B2            ; Padding
F $Y$SPEC,Q     ; Descriptor pointing to the
E               ; System specific field (I-255) : B
```

++  
Define symbols related to the system specific field (mini-message)  
contained in the DAP message header.  
--

\$STRUCT DAP,SSPDEF      System specific field  
F ,L.32      Position to system specific section  
F SSP\_MENU,W      of DAP control block  
V <#      System specific menu field (EX-5) : B  
SSP\_CAP      Menu of fields to follow:  
SSP\_FLG      Extended system capabilities  
TMPT\$,14      Extended flags  
>      Reserved  
K SSP\_MEN\_I,<-      Define SSP\_MENU options that are invalid:  
<DAP\$M\_TMP1\$>!-      Reserved  
0>      Define SSP\_MENU options unsupported by VAX:  
K SSP\_MEN\_U,<-      Padding  
0>      SSP\_FLG,L      System specific flags field (EX-5) : B  
F      Meaning:  
V <#      Load image modifier for open function  
LOAD      Reserved  
TMP1\$,31      Reserved  
>      Define SSP\_FLG options that are invalid:  
K SSP\_FLG\_I,<-      Reserved  
<DAP\$M\_TMP1\$>!-      Define SSP\_FLG options unsupported by VAX:  
0>      K SSP\_FLG\_U,<-      Padding  
0>      F SSP\_CAP,L      System specific capabilities field (EX-5) : B  
V <#      Partner node supports:  
LOADIM      Load image function  
,31      Reserved  
>      Define SSP\_CAP options supported by VAX:  
K SSP\_CAP\_V,<-      LOADIM  
<DAP\$M\_LOADIM>!-      F ,L.1      Spare  
0>  
E .L.1

++  
: Define symbols related to the Configuration message (TYPE=1).  
--

```

$STRUCT DAP,CNFDEF      : DAP Configuration message

M 1
F ,L.16                  : Position to message operand section
                            of DAP control block

F BUFSIZ,W               : Buffer size field (2) : B
                            (This is DAP buffer size value from partner)

F OSTYPE,B               : Operating system type field (1) : B
K <
    RT11,1
    RSTS,2
    RSX11S,3
    RSX11M,4
    RSX11D,5
    IAS,6
    VAXVMS,7
    TOPS20,8
    TOPS10,9
    RTS8,10
    OS8,11
    RSX11MP,12
    COPOS11,13
    P OS,14
    VAXELAN,15
>
F FILESYS,B              : File system type field (1) : B
K <
    RMS11,1
    RMS20,2
    RMS32,3
    FCS11,4
    RT11FS,5
    NO FS,6
    TOPS20FS,7
    TOPS10FS,8
    OS8FS,9
    RMS32S,10
>
F VERNUM,B                : DAP version number field (1) : B
K VERNUM_V,7              : Value for VAX/VMS V4.0
F ECONUM,B                : ECO version number field (1) : B
K ECONUM_V,0              : Value for VAX/VMS V4.0
F USRNUM,B                : User protocol version number field (1) : B
K USRNUM_V,0              : Value for VAX/VMS V4.0
F DECVER,B                : DEC software version number field (1) : B
K DECVER_V,4              : Value for VAX/VMS V4.0
F USRVER,B                : User software version number field (1) : B
K USRVER_V,0              : Value for VAX/VMS V4.0
F ,B,3
F ,L,13
P {
F ,L,10                  : Position to Configuration message save section

```

```

F SYSCAP,0
V <
  FILALL
  SEQORG
  RELORG
  1
  EXTEND
  SEQFIL
  RANRRN
  RANVBN
  RANKEY
  1
  RANRFA
  IDXORG
  SWMODE
  APPEND
  SUBMIT
  1
  MDS
  DISPLAY
  MSGBLK
  UNRBLK
  BIGBLK
  DAPCRC
  KEYXAB
  ALLXAB
  SUMXAB
  DIRECTORY
  TIMXAB
  PROXAB
  1
  FOPSP
  FOPSCF
  FOPDLT
  >
  V <
    .32
    1
    SEQRAC
    1
    BITOPT
    WARNING
    RENAME
    WILDCARD
    GNGOPT
    NAMMSG
    SEGMSG
    CHGATTCLS
    CHGTIMCLS
    CHGPROCLS
    CHGNAMCLS
    MODATTCRE
    : System capabilities field (EX-12) : BM
    : Partner node supports:
    : Allocation of space at file creation
    : Sequential file organization
    : Relative file organization
    : Reserved for HSHORG
    : Manual file extension
    : Sequential file access (file transfer mode)
    : Random access by relative record number
    : Random access by virtual block number
    : Random access by key value
    : Reserved for RANHSH
    : Random access by record file address
    : Multi-keyed indexed file organization
    : Dynamic switching of access modes
    : Append records to end-of-file
    : Command file submission/execution
    : Reserved for COMPRESS (data compression)
    : Multiple data streams per file
    : Display of file attributes on request
    : Blocking of DAP messages up to response
      using a 1-byte length field (LENGTH)
    : Unrestricted blocking of DAP messages
    : Blocking of DAP messages up to response
      using a 2-byte length field (LEN256,LENGTH)
    : DAP message CRC checksum
    : Key Definition XAB message
    : Allocation XAB message
    : Summary XAB message
    : Directory list operation
    : Date and Time XAB message
    : File Protection XAB message
    : Reserved for ACLXAB
    : Spool file on close FOP option
    : Execute command file on close FOP option
    : Delete file on close FOP option
    : Partner node supports:
    : (skip over bits defined above)
    : Reserved for DFTFIL (default file spec)
    : Sequential record access
    : Reserved for RECOVERY
    : Bit count option in the FLAGS field
    : Warning Status message and associated error
      recovery message exchange
    : File rename operation
    : Wildcard operations (excluding directory)
    : Go/Nogo option in the ACCOPT field
    : Name message
    : Segmented DAP messages
    : Changing file attributes on close via ATT msg
    : Changing file attributes on close via TIM msg
    : Changing file attributes on close via PRO msg
    : Changing file attributes on close via NAM msg
      (i.e., rename of file)
    : Modified attributes returned on create
  
```

NAM3PART

CHGATTRN  
CHGTIMREN  
CHGPROREN  
BLKCNT  
OCTALVER

.11

K SYSCAP1 V,-  
<^XEFF67DF7>K SYSCAP2 V,-  
<^X0000T962>

E

3-part Name message format in DISPLAY field  
of both Access and Control messages  
Changing file attributes on rename via ATT msg  
Changing file attributes on rename via TIM msg  
Changing file attributes on rename via PRO msg  
BLKCNT field in Control message  
Octal version numbers only in file specs  
(bit is valid only for DAP V7.0 or later)  
Reserved

Define supported SYSCAP options (bits 00-31):  
FILALL, SEQORG, RELORG, EXTEND, SEQFIL,  
RANRRN, RANVBN, RANKEY, RANRFA, IDXORG, SWMODE,  
APPEND, SUBMIT, DISPLAY, MSGBLK, BIGBLK,  
DAPCRC, KEYXAB, ALLXAB, SUMXAB, DIRECTORY,  
TIMXAB, PROXAB, FOPSPL, FOPSCF, FOPDLT  
Define supported SYSCAP options (bits 32-63):  
SEQRAC, RENAME, WILDCARD, NAMMSG, CHGTIMCLS,  
CHGPROCLS

;;+  
;; Define symbols related to the Attributes message (TYPE=2).  
;:-

\$STRUCT DAP,ATTDEF

F ,L,16

F ATTMENU,L

V <M  
DATATYPE

ORG

RFM

RAT

BLS

MRS

ALQ1

BKS

FSZ

MRN

RUNSYS

DEQ1

FOP1

BSZ

DEV

TMP1\$,1

LRL

HBK

EBK

FFB

SBN

TMP2\$,11

>

K ATTMENU\_I,<-  
<DAPSM\_TMP1\$>|-  
<DAPSM\_TMP2\$>|-  
0>

K ATTMENU\_U,<-  
0>

F DATATYPE,B

V <M

ASCII

IMAGE

TMP1\$,1

CMPFMT

EXEC

PRIV

TMP2\$,1

ZERO

>

K DATATYP\_I,<-  
<DAPSM\_TMP1\$>|-  
0>

K DATATYP\_U,<-  
<DAPSM\_CMPFMT>|-

;; DAP Attributes message

Position to message operand section  
of DAP control block  
Attributes menu field (EX-6) : BM  
Menu of fields to follow:  
Data type  
File organization  
Record format  
Record attributes  
Block size  
Maximum record size  
Allocation quantity  
Bucket size  
Fixed control area size  
Maximum record number  
Run-time system identification  
Default extension quantity  
File options  
Byte size field  
Device characteristics  
Reserved for SDC  
Longest record length  
Highest virtual block number  
End-of-file block number  
First free byte in end-of-file block  
Starting logical block number  
Reserved

Define ATTMENU options that are invalid:

Reserved  
Reserved

Define ATTMENU options unsupported by VAX:

Data type field (EX-2) : BM  
Define offsets and masks:  
Data in ASCII format  
Data in IMAGE format  
Reserved for EBCDIC  
Compressed format  
File contains executable code  
File contains privileged code  
Reserved (ignore if received)  
(this was attributes match flag in DAP V4.1)  
Zero file on erase file operation

Define DATATYPE options that are invalid:

Reserved

Define DATATYPE options unsupported by VAX:

CMPFMT

```

<DAPSM_ZERO>!-          : ZERO
 0>                      Define default DATATYPE value
 K DATATYP D,<-          IMAGE
 <DAPSM_IMAGE>!-          :
 0>                      File organization field (1) : B
 F ORG,B                  File organization:
 K <                      Sequential
   SEQ,0                   Relative
   REL,16                  Indexed
   IDX,32                  (48) reserved for hash
 >
 K ORG_D,DAPSK_SEQ
 F RFM,B                  Define default ORG value
 K <                      Record format field (1) : B
   UDF,0                   Record format:
   FIX,1                   Undefined
   VAR,2                   Fixed length
   VFC,3                   Variable length
   STM,4                   Variable length with fixed control
   STMLF,5                 Stream ASCII
   STMCR,6                 Stream LF
   STMCR,6                 Stream CR
 >
 K RFM_D,DAPSK_FIX
 F RAT,B                  Define default RFM value
 V <M                     Record attributes field (EX-3) : BM
   FTN                     Meaning:
   CR                      Fortran carriage control
   PRN                     Implied (LF-Record-CR) carriage control
   BLK                     Print file format
   EMBEDDED                Records do not cross block boundaries
   TMP1S,1                 Records have embedded control characters
   Reserved                Reserved
   LSA                      Line sequenced ASCII
   MACY11                  MACY11 format
 >
 K RAT_I,<-               Define RAT options that are invalid:
 <DAPSM_TMP1S>!-          Reserved
 0>
 K RAT_U,<-               Define RAT options unsupported by VAX:
 <DAPSM_LSA>!-          LSA
 <DAPSM_MACY11>!-          MACY11
 0>
 K RAT_D,<-               Define default RAT value
 <DAPSM_EMBEDDED>!-          EMBEDDED
 0>

 F BLS,W                  ***** No default value is stated in the
 K BLS_D,512                ***** DAP spec although some systems
 F MRS_W                  ***** treat EMBEDDED as the default
 F ALQf,L                 Block size field (2) : B
 F BKS,B                  Define default BLS value
 F FSZ,B                  Maximum record size field (2) : B
 F BSZ,B                  Allocation quantity field (I-5) : B
 F BSZ,B                  Bucket size field (1) : B
 F BSZ,B                  Fixed control area size field (1) : B
 F BSZ,B                  Byte size field (1) : B
 K BSZ_D,8                 Define default BSZ value

```

```

F ,B
F DEQ1,W
F ,B,2
F MRN,L
F RUN$YS,Q

F FOP1,L
V <M
  RWO
  RWC
  TMP1$,1
  POS
  DLK
  DIR
  FLK
  CTG
  SUP
  NEF
  TMP
  TMD
  TMP2$,1
  DMO
  WCK
  RCK
  CIF
  TMP3$,1
  SQO
  MXV
  SPL
  SCF
  DLT

  CBT
  TMP4$,1
  DFW
  TEF
  OFP
  TMP5$,4
  >
K FOP_I,<-
  <DAPSM_TMP1$>!--
  <DAPSM_TMP2$>!--
  <DAPSM_TMP3$>!--
  <DAPSM_TMP4$>!--
  <DAPSM_TMP5$>!--
  0>
K FOP_U,<-
  <DAPSM_DMO>!--
  0>
F DEV,L
V <
  DEVREC
  DEVCL

```

: Padding  
 : Default extension quantity field (2) : B  
 : Padding  
 : Maximum record number field (I-5) : B  
 : Descriptor pointing to the  
 : Run-time system field (I-40) : A  
 : File options field (EX-6) : BM  
 : Options:  
 : Rewind magtape on open  
 : Rewind magtape on close  
 : Reserved  
 : Position magtape past last created file  
 : Do not lock file if improperly closed  
 : Directory file  
 : File locked  
 : Contiguous space allocation  
 : Supersede existing file on create  
 : Inhibit positioning magtape to end-of-file  
 : Create temporary file  
 : Create temporary file and mark for delete  
 : Reserved  
 : Dismount magtape on close  
 : Enable write checking  
 : Enable read checking  
 : Create if no file present else open file  
 : Reserved for LKO  
 : Sequential access only  
 : Maximize version number  
 : Spool file on close  
 : Submit command file on close  
 : Delete file on close  
 : (used stand-alone or as a suboption to  
 : SCF or SPL)  
 : Contiguous-best-try space allocation  
 : Reserved for WAT  
 : Deferred write (REL and IDX files)  
 : Truncate at EOF on close (SEQ files)  
 : Output file parse  
 : Reserved  
 : Define FOP options that are invalid:  
 : (This is used for both FOP1 and FOP2 fields)  
 : Reserved  
 : Reserved  
 : Reserved  
 : Reserved  
 : Reserved  
 : Define FOP options unsupported by VAX:  
 : (This is used for both FOP1 and FOP2 fields)  
 : DMO  
 : Note: allow DLK, DIR, and FLK  
 : Device characteristics field (EX-6) : BM  
 : Meaning:  
 : Device is record oriented  
 : Carriage control device

DEVTRM  
DEVDIR  
DEVSDI  
DEVSQD  
TMP1\$,.1..M  
DEVFOD  
DEVSHR  
DEVSPL  
DEVMNT  
DEVDMT  
DEVALL  
DEVIDV  
DEVODV  
DEVSWL  
DEVAVL  
DEVELG  
DEVMBX  
DEVRTM  
DEVRND  
DEVRCK  
DEVWCK  
DEVFOR  
DEVNET  
DEVGEN  
TMP2\$,.6..M  
>  
K DEV\_I,<-  
<DAPSM\_TMP1\$>|-  
<DAPSM\_TMP2\$>|-  
0>  
K DEV\_U,<-  
0>  
F ,L,1  
F LRL,W  
F FFB,W  
F HBK,L  
F EBK,L  
F SBN,L  
E

Device is a terminal  
Device is directory structured  
Device is single directory structured  
Seq. block oriented device (e.g., magtape)  
Reserved  
Files oriented device (e.g., disk, magtape)  
Device is sharable  
Device is being spooled  
Device is mounted  
Device is marked for dismount  
Device is allocated  
Device is capable of providing input  
Device is capable of providing output  
Device is software write locked  
Device is available  
Device has error logging enabled  
Device is a mailbox  
Device is realtime in nature  
Device allows random access  
Device has read checking enabled  
Device has write checking enabled  
Device is mounted as foreign (not files-11)  
Network device  
Generic device  
Reserved

Define DEV options that are invalid:  
Reserved  
Reserved

Define DEV options unsupported by VAX:  
Reserved for SDC  
Longest record length field (2) : B  
First free byte in EOF block field (2) : B  
Highest virtual block number field (I-5) : B  
End-of-file block number field (I-5) : B  
Starting logical block number field (I-5) : B

++ Define symbols related to the Access message (TYPE=3).  
--

SSTRUCT DAP,ACCDEF	;	DAP Access message
F ,L,16	;	Position to message operand section of DAP control block
F ACCFUNC,B	;	Access function field (1) : B
K <	;	Access function:
OPEN,1	;	Open a file
CREATE,2	;	Create a file
RENAME,3	;	Rename a file
ERASE,4	;	Erase (delete) a file
DIR LIST,6	;	(5) reserved
SUBMIT,7	;	Return directory list
EXECUTE,8	;	Submit (copy and execute) a command file
LOAD,255	;	Execute a command file
>	;	Load image file--for internal use by FAL
F ACCOPT,B	;	Access options field (EX-5) : BM
V <M	;	Meaning:
NONFATAL	;	I/O errors are not fatal
TMP1\$,2	;	Reserved--used to be STS_STORE and STS_RETRV
RET_CRC	;	Return CRC value with each DAP message
GO_NOGO	;	Go/nogo option
TMP2\$,3	;	Reserved
>	;	Define ACCOPT options that are invalid:
K ACCOPT_I,<-	;	Reserved
<DAPSM_TMP2\$>!-	;	
0>	;	
K ACCOPT_U,<-	;	Define ACCOPT options unsupported by VAX:
<DAPSM_TMP1\$>!-	;	Reserved--was defined in DAP V5.4
<DAPSM_GO_NOGO>!-	;	
0>	;	
F FAC,B	;	File access field (EX-3) : BM
V <M	;	Access for:
PUT	;	Put record
GET	;	Get record
DEL	;	Delete record
UPD	;	Update record
TRN	;	Truncate file
BIO	;	Block I/O operations only
BRO	;	Mixed record and block I/O operations
APP	;	Append record
>	;	Define FAC options that are invalid:
K FAC_I,<-	;	
0>	;	
K FAC_U,<-	;	Define FAC options unsupported by VAX:
0>	;	Note: allow APP
K FAC_D,<-	;	Define default FAC value
<DAPSM_GET>!-	;	GET
0>	;	
F SHR,B	;	File sharing field (EX-3) : BM
V <M	;	Shared access for:
SHRPUT	;	Put record

SHRGET  
SHRDEL  
SHRUPD  
MSE  
UPI  
NIL  
TMP1S,1  
>  
K SHR\_I,<-  
<DAPSM\_TMP1S>!-  
0>  
K SHR\_U,<-  
<DAPSM\_MSE>!-  
0>  
K SHR\_D,<-  
0>  
F FILESPEC,0  
F DISPLAY1,W  
V <M  
DSP\_ATT  
DSP\_KEY  
DSP\_ALL  
DSP\_SUM  
DSP\_TIM  
DSP\_PRO  
TMP1S,2  
DSP\_NAM  
DSP\_3NAM  
TMP2S,6  
>  
K DISPLAY\_I,<-  
<DAPSM\_TMP1S>!-  
<DAPSM\_TMP2S>!-  
0>  
K DISPLAY\_U,<-  
<DAPSM\_DSP\_3NAM>!-  
0>  
F <sup>W</sup> PASSWORD,0  
F ,L,10  
E

Get record  
Delete record  
Update record  
Multiple record streams enabled  
User provided interlocking  
No shared access allowed  
Reserved

Define SHR options that are invalid:  
Reserved

Define SHR options unsupported by VAX:  
MSE

Define default SHR value  
\*\*\*\*\* This is contrary to the DAP spec  
\*\*\*\*\* which says that DAPSM\_GET is the default  
Descriptor pointing to the  
File specification field (I-255) : A  
Display attributes field (EX-4) : BM  
Return the following:  
Attributes message  
Key Definition Attributes message  
Allocation Attributes message  
Summary Attributes message  
Date and Time Attributes message  
Protection Attributes message  
Reserved  
Reserved for ACL Attributes message  
Name message  
3-part Name message  
Reserved

Define DISPLAY options that are invalid:  
(This is used for both DISPLAY1 and DISPLAY2)  
Reserved  
Reserved

Define DISPLAY options unsupported by VAX:  
(This is used for both DISPLAY1 and DISPLAY2)  
3-Part Name message

Padding  
Descriptor pointing to the  
Password field (I-40) : B  
Spare

++  
Define symbols related to the Control message (TYPE=4).  
--

```

$STRUCT DAP,CTLDEF      : DAP Control message
F ,L,16                  : Position to message operand section
F CTLFUNC,B               : of DAP control block
K <                      : Control function field (1) : B
  GET READ,1              : Control function:
  CONNECT,2               :   Get record or read block
  UPDATE,3                :   Establish data stream
  PUT WRITE,4              :   Update record
  DELETE,5                :   Put record or write block
  REWIND,6                :   Delete record
  TRUNCATE,7              :   Rewind file
                           :   Truncate sequential file
                           :   (8) reserved for modify file attributes
                           :   Release locked record
                           :   Free all locked records
                           :   Extend file (beginning message of sequence)
                           :   Flush all records
                           :   (13) reserved for next volume processing
                           :   Find record
                           :   Extend file (ending message of sequence)
                           :   Display file attributes
                           :   Space file forward
                           :   Space file backward
                           :   (19) reserved for checkpoint file
                           :   (20) reserved for recovery get
                           :   (21) reserved for recovery put
>
F B,3                    : Padding
F CTLMENU,W              : Control menu field (EX-4) : BM
V <M
  RAC
  KEY
  KRF
  ROP
  TMP1$,1
  DISPLAY2
  BLKCNT
  TMP2$,9
>
K CTLMENU,I,<-           : Define CTLMENU options that are invalid:
  <DAPSM_TMP1$>!-         :   Reserved
  <DAPSM_TMP2$>!-         :   Reserved
  0>
K CTLMENU,U,<-           : Define CTLMENU options unsupported by VAX:
  <DAPSM_BLKCNT>!-        :   BLKCNT
  0>
F RAC,B                  : Record access field (1) : B
K <
  SEQ_ACC,0                : Record access type:
  KEY_ACC,1                :   Sequential record access
  RFA_ACC,2                :   Random access by key value or record number
                           :   Random access by RFA

```

```

SEQFILE3 : Sequential file transfer mode
BLKVBN4 : Block I/O access by VBN
BLKFILE5 : Block I/O file transfer mode
>
K RAC_D,DAPSK_SEQ_ACC : Define default RAC value
F KRF,B : Key of reference field (1) : B
F KEY,O : Descriptor pointing to the
           Key field (I-255) : B
F ROP_L : Record options field (EX-6) : BM
V <M : Meaning:
       EOF
       FDL
       UIF
       TMP1$,.1
       LOA
       ULK
       TPT
       RAH
       UBH
       KGE
       KGT
       NLK
       RLK
       ROPBIO
       LIM
       NXR
       ROPWAT
       RRL
       REA
       TMP2$,.13
>
K ROP_I,<- : Define ROP options that are invalid:
<DAPSM_TMP1$>!- : Reserved
<DAPSM_TMP2$>!- : Reserved
O>
K ROP_U,<- : Define ROP options unsupported by VAX:
O>
F DISPLAY2,W : Display attributes field (EX-4) : BM
                (see DISPLAY1 field of Access message
                 for bit definitions)
F BLKCNT,B : Block count field
F ,B : Padding
F ,L,10 : Spare
E

```

++  
Define symbols related to the Continue Transfer message (TYPE=5).  
--

```
$STRUCT DAP,CONDEF      DAP Continue Transfer message
F ,L,16                  Position to message operand section
F CONFUNC,B              of DAP control block
K <                     Continue transfer function field (1) : B
  RETRY,1                 Recovery action:
  SKIP REC,2              Try access function again
  ABORT,3                 Skip record in error and continue
  RESUME,4                Abort request
  QUIT,5                  Resume operation
  >                      Terminate file processing
  F ,B,3
  F ,L,15
E
```

Padding  
Spare

;++  
; Define symbols related to the Acknowledge message (TYPE=6).  
;--

STRUCT DAP,ACKDEF ; DAP Acknowledge message  
F ,L,16 ; Position to message operand section  
F ,L,16 ; of DAP control block  
E ; Spare

++  
Define symbols related to the Access Complete message (TYPE=7).  
--

```
$STRUCT DAP,CMPDEF      ; DAP Access Complete message
F .L.16
F CMPFUNC,B
K <
  CLOSE,1
  RESPONSE,2
  RESET,3
  DISCONN,4
  SKIP_FILE,5
  CHANGE_B,6
  CHANGE_E,7
  TERMINATE,8
  >
F .B
F CHECK,W
F FOP2,L
F .L.14
E

; Position to message operand section
; of DAP control block
; Access complete function field (1) : B
; Access complete function:
;   Close file
;   Response to partner's CMPFUNC request
;   Close file and restore it to initial state
;   (this used to be named PURGE)
;   Disconnect record stream
;   Skip to next file (i.e., close this file
;   and open next file)
;   Close file and change its file attributes
;   (beginning message of sequence)
;   Close file and change its file attributes
;   (ending message of sequence)
;   Terminate (abort) operation and re-initialize
; Padding
; CRC checksum field (2) : B
; File options field (EX-6) : BM
; (see FOP1 field of Attributes message
; for bit definitions)
; Spare
```

```
++  
Define symbols related to the Data message (TYPE=8).  
--
```

```
$STRUCT DAP,DATDEF      ; DAP Data message  
F ,L,16                  ; Position to message operand section  
F RECNUM1,L              ; of DAP control block  
F FILEDATA,Q              ; Record number field (I-8) : B  
F ,L,13                  ; Descriptor pointing to the  
E                          ; File data field (rest-of-message) : B  
                           ; Spare
```

++  
Define symbols related to the Status message (TYPE=9).  
--

```
STRUCT DAP,STSDEF : DAP Status message
F ,L,16
F STSCODE,W
  V <M
    MICCODE,12
    MACCODE,4
  >
  K <S
    PENDING,0
    SUCCESS,1
    UNSUPPORT,2
    FILE_OPEN,4
    FILE_XFER,5
    WARNING,6
    FILE_CLOS,7
    FORMAT,8
    INVALID,9
    MSG_SYNC,10
  >
  F RFA,W,3
  F RECNUM2,L
  F STV,L
  F STX,Q
  F ,L,10
E

Position to message operand section
of DAP control block
DAP status code field (2) : B
Subfields:
  Micro status code
  Macro status code

MACCODE field status code classes:
  Operation in progress
  Operation completed successfully
  DAP implementation does not support request
  (3) reserved
  Error related to opening a file
  Error encountered while file was open
  (i.e., during record access)
  Warning error condition
  Error related to closing a file
  Parse error caused by incorrect format
  Invalid DAP field value
  DAP message received out-of-order

Record file address field (I-8) : B
Record number field (I-8) : B
Secondary status field (I-8) : B
Descriptor pointing to the
Secondary status text field (I-255) : A
Spare
```

++  
 Define symbols related to the Key Definition Attributes message (TYPE=10).  
 --

```

$STRUCT DAP,KEYDEF      : DAP key definition Attributes message
F ,L,16                  Position to message operand section
F KEYMENU,L              of DAP control block
V <M                      Key definition menu field (EX-6) : BM
  FLG
  DFL
  IFL
  NSG
  REF
  KNM
  NUL
  IAN
  LAN
  DAN
  DTP
  RVB
  TMP1$,1
  DVB
  DBS
  IBS
  LVL
  TKS
  MRL
  TMP2$,13
  >
  K KEYMENU_I,<-
    <DAPSM_TMP1$>!-
    <DAPSM_TMP2$>!-
    0>
  K KEYMENU_U,<-
    0>
F DFL,W
F IFL,W
F FLG,B
V <M
  DUP
  CHG
  NUL CHR
  TMPT$,5
  >
  K FLG_I,<-
    <DAPSM_TMP1$>!-
    0>
  K FLG_U,<-
    0>
F NSG,B
F POS TMP,W
  S SIZ TMP,0,B
F POS,0,8
  S POS0,0,W

```

:
 : DAP key definition Attributes message

:
 : Position to message operand section

:
 : of DAP control block

:
 : Key definition menu field (EX-6) : BM

:
 : Menu of fields to follow:

:
 : FLG

:
 : DFL

:
 : IFL

:
 : NSG, POS, SIZ

:
 : REF

:
 : KNM

:
 : NUL

:
 : IAN

:
 : LAN

:
 : DAN

:
 : DTP

:
 : RVB

:
 : Reserved for HAL

:
 : DVB

:
 : DBS

:
 : IBS

:
 : LVL

:
 : TKS

:
 : MRL

:
 : Reserved

:
 : Define KEYMENU options that are invalid:

:
 : Reserved

:
 : Reserved

:
 : Define KEYMENU options unsupported by VAX:

:
 : Data bucket fill quantity field (2) : B

:
 : Index bucket fill quantity field (2) : B

:
 : Key options field (EX-3) : BM

:
 : Meaning:

:
 : Duplicate key values allowed

:
 : Key field may change on update (alt key)

:
 : Null key character defined (alt key)

:
 : Reserved

:
 : Define key options (FLG) that are invalid:

:
 : Reserved

:
 : Define key options (FLG) unsupported by VAX:

:
 : Number of key segments field (1) : B

:
 : Temporary work space for POS field processing

:
 : Temporary work space for SIZ field processing

:
 : Key segment position field (2) : B

:
 : Segment 0

```

S POS1,2,W          : Segment 1
S POS2,4,W          : Segment 2
S POS3,6,W          : Segment 3
S POS4,8,W          : Segment 4
S POS5,10,W         : Segment 5
S POS6,12,W         : Segment 6
S POS7,14,W         : Segment 7
F SIZ,B             : Key segment size field (1) : B
S SIZ0,0,B          : Segment 0
S SIZ1,1,B          : Segment 1
S SIZ2,2,B          : Segment 2
S SIZ3,3,B          : Segment 3
S SIZ4,4,B          : Segment 4
S SIZ5,5,B          : Segment 5
S SIZ6,6,B          : Segment 6
S SIZ7,7,B          : Segment 7
F KNM,Q             : Descriptor pointing to the
                       Key name field (I-40) : A
F REF,B              : Key of reference field (1) : B
F NUL,B              : Null key character field (1) : B
F IAN,B              : Index area number field (1) : B
F LAN,B              : Lowest level index area number field (1) : B
F DAN,B              : Data area number field (1) : B
F DTP,B              : Key data type field (1) : B
K <
  STG,0              : Data type:
  IN2,1              : String
  BN2,2              : Signed 2-byte integer
  IN4,3              : Unsigned 2-byte integer (binary)
  BN4,4              : Signed 4-byte integer
  PAC,5              : Unsigned 4-byte integer (binary)
  IN8,6              : Packed decimal (0-31 digits plus sign)
  BN8,7              : Signed 8-byte integer
  >
  K DTP_D,DAPSK_STG : Unsigned 8-byte integer (binary)
  Define default DTP value
  F MRL,W             : Minimum record length to contain key field (2) : B
  F RVB,L              : Root bucket start VBN field (I-8) : B
  F DVB,L              : First data bucket start VBN field (I-8) : B
  F DBS,B              : Data bucket fill size field (1) : B
  F IBS,B              : Index bucket fill size field (1) : B
  F LVL,B              : Level of root buckets field (1) : B
  F TKS,B              : Total key size field (1) : B
E

```

++  
 ; Define symbols related to the Allocation Attributes message (TYPE=11).  
 ;--

\$STRUCT DAP,ALLDEF ; DAP Allocation Attributes message  
 F ,L,16 ; Position to message operand section  
 F ALLMENU,W ; of DAP control block  
 V <M ; Allocation menu field (EX-6) : BM  
 Menu of fields to follow:  
 VOL  
 ALN  
 AOP  
 LOC  
 TMP1\$,1  
 ALQ2  
 AID  
 BKZ  
 DEQ2  
 TMP2\$,7  
 >  
 K ALLMENU\_I,<- ; Define ALLMENU options that are invalid:  
 <DAPSM\_TMP1\$>!- ; Reserved  
 <DAPSM\_TMP2\$>!- ; Reserved  
 0>  
 K ALLMENU\_U,<- ; Define ALLMENU options unsupported by VAX:  
 0>  
 F VOL,W ; Relative volume number field (2) : B  
 F ALN,B ; Alignment options field (EX-4) : BM  
 K < ; Alignment types:  
 ANY,0 ; Any allocation placement is ok  
 CYL,1 ; Align on cylinder boundary  
 LBN,2 ; Align on specified logical block  
 VBN,3 ; Allocate near specified virtual block  
 RFI,4 ; Allocate near specified related file  
 >  
 F AOP,B ; Allocation options field (EX-4) : BM  
 V <M ; Options:  
 HRD  
 CTG2  
 CBT2  
 ONC  
 TMP1\$,4  
 >  
 K AOP\_I,<- ; Define AOP options that are invalid:  
 <DAPSM\_TMP1\$>!- ; Reserved  
 0>  
 K AOP\_U,<- ; Define AOP options unsupported by VAX:  
 0>  
 F ,B,2 ; Padding  
 F LOC\_L ; Starting location field (I-8) : B  
 F ALQ2\_L ; Allocation quantity field (I-5) : B  
 F AID,B ; Area identification field (1) : B  
 F BKZ,B ; Bucket size field (1) : B  
 F DEQ2,W ; Default extension quantity field (2) : B  
 F ,L,11 ; Spare

DAPDEF.MDL:1

16-SEP-1984 16:38:17.80 <sup>M 16</sup> Page 28

E

:

++  
Define symbols related to the Summary Attributes message (TYPE=12).  
--

STRUCT DAP,SUMDEF

F .L.16

F SUMENU,W

V <M

NOK

NOA

NOR

PVN

TMP1\$,12

>

K SUMENU\_I,<-  
<DAP\$M\_TMP1\$>!-  
0>

K SUMENU\_U,<-  
0>

F PVN,W

F NOK,B

F NOA,B

F NOR,B

F .B

F .L.14

E

; DAP summary Attributes message

; Position to message operand section  
; of DAP control block

; Summary menu field (EX-6) : BM

; Menu of fields to follow:

; NOK

; NOA

; NOR

; PVN

; Reserved

; Define SUMENU options that are invalid:

; Reserved

; Define SUMENU options unsupported by VAX:

; Note: allow NOR

; Prologue version number field (1) : B

; Number of keys field (1) : B

; Number of allocation areas field (1) : B

; Number of record descriptors field (1) : B

; Padding

; Spare

++  
: Define symbols related to the Date and Time Attributes message (TYPE=13).  
--

```
$STRUCT DAP,TIMDEF      : DAP date and time Attributes message
F .L.16
F TIMENU,W
  V <M
    CDT
    RDT
    EDT
    RVN
    BDT
    PDT
    ADT
    TMP1$,.9
  >
  K TIMENU_I,<-
    <DAPSM_TMP1$>!-
  0>
  K TIMENU_U,<-
  0>
F RVN,W
F .L
F CDT,Q
F RDT,Q
F EDT,Q
F BDT,Q
F PDT,Q
F ADT,Q
F .L.2
E
```

Position to message operand section  
of DAP control block  
Date and time menu field (EX-6) : BM  
Menu of fields to follow:  
CDT  
RDT  
EDT  
RVN  
BDT  
PDT  
ADT  
Reserved

Define TIMENU options that are invalid:  
Reserved

Define TIMENU options unsupported by VAX:  
Note: allow PDT and ADT  
Revision number field (2) : B  
Padding  
Creation date and time field (18) : A  
(stored in DAP control block as a  
64-bit time value per VMS convention)  
Revision date and time field (18) : A  
(stored in DAP control block as a  
64-bit time value per VMS convention)  
Expiration date and time field (18) : A  
(stored in DAP control block as a  
64-bit time value per VMS convention)  
Backup date and time field (18) : A  
(stored in DAP control block as a  
64-bit time value per VMS convention)  
Physical creation date and time field (18) : A  
(stored in DAP control block as a  
64-bit time value per VMS convention)  
Accessed date and time field (18) : A  
(stored in DAP control block as a  
64-bit time value per VMS convention)  
Spare

++  
Define symbols related to the Protection Attributes message (TYPE=14).  
--

\$STRUCT DAP,PRODEF : DAP protection Attributes message  
F ,L.16  
F PROMENU,W : Position to message operand section  
V <M of DAP control block  
OWNER Protection menu field (EX-6) : BM  
PROSYS  
PROOWN  
PROGRP  
PROWLD  
TMP1\$,.11  
Menu of fields to follow:  
OWNER  
PROSYS  
PROOWN  
PROGRP  
PROWLD  
Reserved  
K PROMENU\_I,<- Define PROMENU options that are invalid:  
<DAPSM\_TMP1\$>!-  
0>  
K PROMENU\_U,<- Define PROMENU options unsupported by VAX:  
0>  
F ,W,3 Padding  
F OWNER,Q Descriptor pointing to the  
File owner field (I-40) : A  
F PROSYS,W System protection field (EX-3) : BM  
V <M Meaning:  
RED\_ACC Deny read access  
WRT\_ACC Deny write access  
EXE\_ACC Deny execute access  
DLT\_ACC Deny delete access  
APP\_ACC Deny append access  
DIR\_ACC Deny directory access  
UPD\_ACC Deny update access  
CHG\_ACC Deny change protection access  
EXT\_ACC Deny extend access  
TMPTS,7 Reserved  
K PROTECT\_I,<- Define protection options that are invalid:  
<DAPSM\_TMP1\$>!-  
0>  
K PROTECT\_U,<- Define protection options unsupported by VAX:  
0>  
F PROOWN,W This mask applies to PROSYS, PROOWN, PROGRP,  
F PROGRP,W and PROWLD fields  
F PROWLD,W Define protection options unsupported by VAX:  
F ,L.10 This mask applies to PROSYS, PROOWN, PROGRP,  
Note: allow APP\_ACC, DIR\_ACC, UPD\_ACC,  
CHG\_ACC, and EXT\_ACC  
Owner protection field (EX-3) : BM  
Group protection field (EX-3) : BM  
World protection field (EX-3) : BM  
Spare

++  
: Define symbols related to the Name Attributes message (TYPE=15).  
:--

\$STRUCT DAP,NAMDEF : DAP name Attributes message  
F ,L,16 Position to message operand section  
F NAMETYPE,B of DAP control block  
V <M Name type field (EX-3) : BM  
FILSPEC  
FILNAME  
DIRNAME  
VOLNAME  
DFTSPEC  
TMP1\$,1  
TMP2\$,2  
Type:  
Primary file specification  
File name  
Directory name  
Volume or structure name  
Default file specification  
Reserved for RELSPEC  
Reserved  
>  
F ,B,3 Padding  
K NAMETYP I,<- Define NAMETYPE options that are invalid:  
<DAP\$M\_TMP1\$>!- Reserved  
<DAP\$M\_TMP2\$>!- Reserved  
0>  
K NAMETYP U,<- Define NAMETYPE options unsupported by VAX:  
<DAP\$M\_DFTSPEC>!- DFTSPEC  
0>  
F NAMESPEC,0 Descriptor pointing to the  
F ,L,13 Name field (I-255) : A  
E Spare

++  
Define symbols related to DAP message CRC checksum computation.  
The CRC polynomial function (order 16) used is:

---

$x^{16} + x^{15} + x^{13} + x^7 + x^4 + x^2 + x^1 + 1$

STRUCT DAP,CRCDEF : DAP message CRC checksum symbol definitions

K CRC\_INIT,<^X0000FFFF> : Initial CRC value  
K CRC\_POLY,<^X0000E905> : CRC polynomial representation used as  
: input to LIB\$CRC\_TABLE to generate  
: the CRC polynomial table below:  
K CRC\_TBL0,<^X00000000> : Table entry 0  
K CRC\_TBL1,<^X000053E3> : Table entry 1  
K CRC\_TBL2,<^X0000A7C6> : Table entry 2  
K CRC\_TBL3,<^X0000F425> : Table entry 3  
K CRC\_TBL4,<^X00009D87> : Table entry 4  
K CRC\_TBL5,<^X0000CE64> : Table entry 5  
K CRC\_TBL6,<^X00003A41> : Table entry 6  
K CRC\_TBL7,<^X000069A2> : Table entry 7  
K CRC\_TBL8,<^X0000E905> : Table entry 8  
K CRC\_TBL9,<^X0000BAE6> : Table entry 9  
K CRC\_TBLA,<^X00004EC3> : Table entry 10  
K CRC\_TBLB,<^X00001D20> : Table entry 11  
K CRC\_TBLC,<^X00007482> : Table entry 12  
K CRC\_TBLD,<^X00002761> : Table entry 13  
K CRC\_TBLE,<^X0000D344> : Table entry 14  
K CRC\_TBLF,<^X000080A7> : Table entry 15

E

++  
: \$DAPFIDDEF defines DAP field identification code symbols.  
: These are used to identify a field in a DAP Status message.  
--

\$STRUCT DAP,FIDDEF : DAP field ID codes

K <,\$  
UNKNOWN,0  
TYPE,8  
>  
K <,\$  
FLAGS,8  
STREAMID,9  
LENGTH,10  
LEN256,11  
BITCNT,12  
SYSPEC,14  
SSP\_MENU,14  
SSP\_CAP,14  
SSP\_FLG,14  
>  
K <,\$  
BUFSIZ,16  
OSTYPE,17  
FILESYS,18  
VERNUM,19  
ECONUM,20  
USRNUM,21  
DECVER,22  
USRVER,23  
SYSCAP,24  
>  
K <,\$  
ATTMENU,16  
DATATYPE,17  
ORG,18  
RFM,19  
RAT,20  
BLS,21  
MRS,22  
ALQ,23  
BKS,24  
FSZ,25  
MRN,26  
RUNSYS,27  
DEQ1,28  
FOP1,29  
BSZ,30  
DEV,31  
LRL,33  
HBK,34  
EBK,35  
: Miscellaneous field codes:  
: Unknown field  
: DAP message type field  
: Message header field codes:  
: DAP message flags field  
: Data stream identification field  
: Length field  
: Length extension field  
: Bit count field  
: (13) reserved  
: System specific field  
: whose subfields use the same code:  
: System specific menu field  
: System specific capabilities field  
: System specific flags field  
: Configuration message field codes:  
: Buffer size field  
: Operating system type field  
: File system type field  
: DAP version number field  
: ECO version number field  
: User protocol version number field  
: DEC software version number field  
: User software version number field  
: System capabilities field  
: Attributes message field codes:  
: Attributes menu field  
: Data type field  
: File organization field  
: Record format field  
: Record attributes field  
: Block size field  
: Maximum record size field  
: Allocation quantity field  
: Bucket size field  
: Fixed control area size field  
: Maximum record number field  
: Run-time system field  
: Default extension quantity field  
: File options field  
: Byte size field  
: Device characteristics field  
: (32) reserved for SDC field  
: Longest record length field  
: Highest virtual block number field  
: End-of-file block number field

```

FFB,36
SBN,37
>
K <,$
ACC FUNC,16
ACCOPT,17
FILESPEC,18
FAC,19
SHR,20
DISPLAY1,21
PASSWORD,22
>
K <,$
CTL FUNC,16
CTL MENU,17
RAC,18
KEY,19
KRF,20
ROP,21
DISPLAY2,23
BLKCNT,24
>
K <,$
CONFUNC,16
>
K <,$
CMP FUNC,16
FUP2,17
CHECK,18
>
K <,$
RECNUM1,16
FILEDATA,17
>
K <,$
SISCODE,16
RFA,18
RECNUM2,19
STV,20
STX,21
>
K <,$
KEY RENU,16
FLG,17
DFL,18
IFL,19
NSG,20
POS,21
POS TMP,21
SIZ,22
SIZ TMP,22
>
First free byte in EOF block field
Starting logical block number field

Access message field codes:
Access function field
Access options field
File specification field
File access field
File sharing field
Display attributes field
Password field

Control message field codes:
Control function field
Control menu field
Record access field
Key field
Key of reference field
Record options field
(22) reserved for HSH field
Display attributes field
Block count field

Continue Transfer message field codes:
Continue transfer function field

Acknowledge message field codes:
none

Access Complete message field codes:
Access complete function field
File options field
CRC Checksum field

Data message field codes:
Record number field
File data field

Status message field codes:
Status code field used for both:
MACCODE,16
MICCODE,17
Record file address field
Record number field
Secondary status value field
Secondary status text field

Key definition attributes message field codes:
Key definition menu field
Key options field
Data bucket fill quantity field
Index bucket fill quantity field
Number of key segments field
Key segment position field
(alias for POS)
Key segment size field
(alias for SIZ)

```

REF.23 : Key of reference field  
KNM.24 : Key name field  
NUL.25 : Null key character field  
IAN.26 : Index area number field  
LAN.27 : Lowest level index area number field  
DAN.28 : Data area number field  
DTP.29 : Key data type field  
RVB.30 : Root bucket start VBN field  
DVB.32 : (31) reserved for HAL field  
DBS.33 : First data bucket start VBN field  
IBS.34 : Data bucket fill size field  
LVL.35 : Index bucket fill size field  
TKS.36 : Level of root buckets field  
MRL.37 : Total key size field  
Minimum record length to contain key field  
>  
K <,\$ : Allocation attributes message field codes:  
ALLMENU.16 : Allocation menu field  
VOL.17 : Relative volume number field  
ALN.18 : Alignment options field  
AOP.19 : Allocation options field  
LOC.20 : Starting location field  
ALQ2.22 : (21) reserved for RFI field  
AID.23 : Allocation quantity field  
BKZ.24 : Area identification field  
DEQ2.25 : Bucket size field  
Default extension quantity field  
>  
K <,\$ : Summary attributes message field codes:  
SUMMENU.16 : Summary menu field  
NOK.17 : Number of keys field  
NOA.18 : Number of allocation areas field  
NOR.19 : Number of record descriptors field  
PVN.20 : Prologue version number field  
>  
K <,\$ : Date and time attributes message field codes:  
TIMENU.16 : Date and time menu field  
CDT.17 : Creation date and time field  
RDT.18 : Revision date and time field  
EDT.19 : Expiration date and time field  
RVN.20 : Revision number field  
BDT.21 : Backup date and time field  
PDT.22 : Physical creation date and time field  
ADT.23 : Accessed date and time field  
>  
K <,\$ : Protection attributes message field codes:  
PROMENU.16 : Protection menu field  
OWNER.17 : File owner field  
PROSYS.18 : System protection field  
PROOWN.19 : Owner protection field  
PROGRP.20 : Group protection field  
PROWLD.21 : World protection field  
>  
K <,\$ : Name message field codes:  
NAMETYPE.16 : Name type field  
NAMESPEC.17 : Name field

E &gt;

:

: End of module

0173 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

SCHFEB  
LTS

SHFDIR  
LTS

SND\$MB  
LTS

SELVOL  
LTS

SNDR  
LTS

TRUNC  
LTS

FAL

FAL  
MAP

DARDEF  
MOL

SMALOC  
LTS

SNOBAD  
LTS

SNTUL  
LTS

WTURN  
LTS

0174 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

## FALBLOXAB LIS